

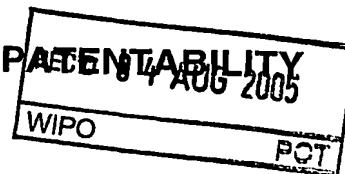
# PATENT COOPERATION TREATY



## PCT

### INTERNATIONAL PRELIMINARY REPORT ON PATENTABILITY

(Chapter II of the Patent Cooperation Treaty)

(PCT Article 36 and Rule 70)



Applicant's or agent's file reference GB030030		<b>FOR FURTHER ACTION</b>		See Form PCT/PEA/416
International application No. PCT/GB2004/000910		International filing date (day/month/year) 04.03.2004	Priority date (day/month/year) 24.04.2003	
International Patent Classification (IPC) or national classification and IPC G06F9/445, G06F9/45				
Applicant INTERNATIONAL BUSINESS MACHINES CORPORATION				
<p>1. This report is the international preliminary examination report, established by this International Preliminary Examining Authority under Article 35 and transmitted to the applicant according to Article 36.</p> <p>2. This REPORT consists of a total of 7 sheets, including this cover sheet.</p> <p>3. This report is also accompanied by ANNEXES, comprising:</p> <p>a. <input checked="" type="checkbox"/> sent to the applicant and to the International Bureau a total of 3 sheets, as follows:</p> <p><input checked="" type="checkbox"/> sheets of the description, claims and/or drawings which have been amended and are the basis of this report and/or sheets containing rectifications authorized by this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions).</p> <p><input type="checkbox"/> sheets which supersede earlier sheets, but which this Authority considers contain an amendment that goes beyond the disclosure in the international application as filed, as indicated in item 4 of Box No. I and the Supplemental Box.</p> <p>b. <input type="checkbox"/> (sent to the International Bureau only) a total of (Indicate type and number of electronic carrier(s)) , containing a sequence listing and/or tables related thereto, in computer readable form only, as indicated in the Supplemental Box Relating to Sequence Listing (see Section 802 of the Administrative Instructions).</p>				
<p>4. This report contains indications relating to the following items:</p> <p><input checked="" type="checkbox"/> Box No. I Basis of the opinion</p> <p><input checked="" type="checkbox"/> Box No. II Priority</p> <p><input type="checkbox"/> Box No. III Non-establishment of opinion with regard to novelty, inventive step and industrial applicability</p> <p><input type="checkbox"/> Box No. IV Lack of unity of invention</p> <p><input checked="" type="checkbox"/> Box No. V Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement</p> <p><input type="checkbox"/> Box No. VI Certain documents cited</p> <p><input type="checkbox"/> Box No. VII Certain defects in the international application</p> <p><input checked="" type="checkbox"/> Box No. VIII Certain observations on the international application</p>				
Date of submission of the demand  22.02.2005		Date of completion of this report  04.08.2005		
Name and mailing address of the international preliminary examining authority:  European Patent Office - P.B. 5818 Patentlaan 2 NL-2280 HV Rijswijk - Pays Bas Tel. +31 70 340 - 2040 Tx: 31 651 epo nl Fax: +31 70 340 - 3016		Authorized Officer  Dieben, M  Telephone No. +31 70 340-4440 		

**INTERNATIONAL PRELIMINARY REPORT  
ON PATENTABILITY**

International application No.  
PCT/GB2004/000910

---

**Box No. I Basis of the report**

---

1. With regard to the **language**, this report is based on the international application in the language in which it was filed, unless otherwise indicated under this item.
- ☐ This report is based on translations from the original language into the following language , which is the language of a translation furnished for the purposes of:
- ☐ international search (under Rules 12.3 and 23.1(b))
  - ☐ publication of the international application (under Rule 12.4)
  - ☐ international preliminary examination (under Rules 55.2 and/or 55.3)
2. With regard to the **elements\*** of the international application, this report is based on *(replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report):*

**Description, Pages**

1-11 as originally filed

**Claims, Numbers**

1-13 received on 22.02.2005 with letter of 22.02.2005

**Drawings, Sheets**

1/4-4/4 as originally filed

- ☐ a sequence listing and/or any related table(s) - see Supplemental Box Relating to Sequence Listing
3. ☐ The amendments have resulted in the cancellation of:
- ☐ the description, pages
  - ☐ the claims, Nos.
  - ☐ the drawings, sheets/figs
  - ☐ the sequence listing (*specify*):
  - ☐ any table(s) related to sequence listing (*specify*):
4. ☐ This report has been established as if (some of) the amendments annexed to this report and listed below had not been made, since they have been considered to go beyond the disclosure as filed, as indicated in the Supplemental Box (Rule 70.2(c)).
- ☐ the description, pages
  - ☐ the claims, Nos.
  - ☐ the drawings, sheets/figs
  - ☐ the sequence listing (*specify*):
  - ☐ any table(s) related to sequence listing (*specify*):

\* If item 4 applies, some or all of these sheets may be marked "superseded."

**INTERNATIONAL PRELIMINARY REPORT  
ON PATENTABILITY**

International application No.  
PCT/GB2004/000910

---

**Box No. II Priority**

---

1. ☐ This report has been established as if no priority had been claimed due to the failure to furnish within the prescribed time limit the requested:
- ☐ copy of the earlier application whose priority has been claimed (Rule 66.7(a)).
  - ☐ translation of the earlier application whose priority has been claimed (Rule 66.7(b)).
2. ☐ This report has been established as if no priority had been claimed due to the fact that the priority claim has been found invalid (Rule 64.1). Thus for the purposes of this report, the international filing date indicated above is considered to be the relevant date.
3. Additional observations, if necessary:
- see separate sheet**

---

**Box No. V Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement**

---

1. Statement
- |                               |             |      |
|-------------------------------|-------------|------|
| Novelty (N)                   | Yes: Claims | 1-13 |
|                               | No: Claims  |      |
| Inventive step (IS)           | Yes: Claims |      |
|                               | No: Claims  | 1-13 |
| Industrial applicability (IA) | Yes: Claims | 1-13 |
|                               | No: Claims  |      |
2. Citations and explanations (Rule 70.7):
- see separate sheet**

---

**Box No. VIII Certain observations on the international application**

---

The following observations on the clarity of the claims, description, and drawings or on the question whether the claims are fully supported by the description, are made:

**see separate sheet**

**Re Item II**

1. The International Searching Authority has not been able to consider the validity of the priority claim because a copy of the earlier application whose priority has been claimed as not available at the time that the search was conducted (Rule 17.1). This opinion has nevertheless been established on the assumption that the relevant date is the claimed priority date.

**Re Item V**

2. The following document is referred to in this communication:

D1: EP0913769

3. The present application does not meet the criteria of Article 33(1) PCT, because the subject-matter of claims 1-13 does not involve an inventive step in the sense of Article 33(3) PCT, for the following reasons:

- 3.01 The subject-matter of independent claim 1 does not involve an inventive step as D1 discloses:

*A data processing method for creating (D1, §0048, "a method for ... packaging a set of class files") an executable file (D1, §0048, "mclass or multi-class file") by combining a plurality of run units (D1, §0048, "...package classes..."), the method comprising the steps of:*

*reading a first run unit (D1, §0049, "step 401, the preprocessor reads and parses each class") to be added to the executable file;*

*locating a first ... data entity set to a first string value (D1, §0049, "....determine ... class file constants, such as strings ...") in the first run unit (D1, §0049, "each class");*

*comparing the first ... data entity with a second data entity set to a second string value the second ... data entity being from the second run unit (D1, §0049, "step 402 ... all duplicate constants determined ...") previously added to the executable file; and*

*adding the first run unit to the executable file ... (D1, §0051, "...step 407, the pre-processor produces a multi-class file...")*

The subject-matter of claim 1 differs from D1 in that claim 1 further defines *adding the first run unit to the executable file but without the first ... data entity in dependence on the result of the comparison* whereas D1 discloses replacing the first data entity with a pointer to a shared symbol table (D1, §0053, "... an element of the new constant type replaces the duplicated element in the reduced pool to direct constant resolution to the shared element in the shared constant pool. Reduction occurs because the replacement element is just a pointer to the actual constant placed in the shared constant pool."). The effect of said difference is that the system of claim 1 allows more compression since there are no pointers left in the code to point to constants in a shared constant table.

The objective technical problem to be solved is how to compress code without the loss of information.

The subject-matter of claim 1 further differs from D1 in that claim 1 further defines that the data entity is non-executable. The effect of said difference is that redundant data entities that are not used in the executable are deleted from the executable file (See also Item VIII).

However the person skilled in the art faced with the problem of how to compress code without the loss of information would realize that the pointers to a shared constant table are not needed in case the data entity is not used in the execution of the program and would therefor not add a pointer to a shared constant table to replace the deleted data entity and arrive at the invention of claim 1 without the need of an inventive step.

3.02 The additional subject-matter of claim 2 does not involve an inventive step as D1 discloses:

*the step of matching matches the first data entity with the second data entity if the first string value and the second string value are identical* (D1, §0049, "step 402 ... all duplicate constants determined ...").

3.03 The subject-matter of apparatus claims 7 and 8 does not involve an inventive step as the objection made to methods claims 1 and 2 apply mutatis mutandis to corresponding apparatus claims 7 and 8.

- 3.04 The subject-matter of claims 3 and 4 differs from D1 in that the method of claims 3 and 4 defines *matching a first non-executable data entity with a second non-executable data entity if the second string value contains the first string value*. The effect of said difference is that in case a non-executable data entity is a substring of another non-executable data entity, the substring non-executable data entity is not added to the executable file. It is a choice of design obvious to the person skilled in the art to define rules as to when a comparison of two string values is considered to be a match. When a match is found it is the objective of the method of D1 to remove one of the matching data entities. In case both string values are identical it is an arbitrary choice obvious to the person skilled in the art to choose either one of them. However when in the presented case the strings are not identical (one is a substring of the other string) it is also a choice of design obvious to the person skilled in the art to define a rule to remove a particular string (in this case the substring). Moreover the technical advantage of not adding a substring is unclear. It appears that not adding a substring may result in a loss of information since the semantics of a substring need not be identical to the semantics of the string.
- 3.05 As far the subject-matter of claim 5 can be understood it appears to differ from D1 in that claim 5 further defines:  
*locating two or more non-executable data entities ...; and creating the first data entity from the located non-executable data entities*  
This newly created first non-executable data entity is added to the executable file. However the created first non-executable data entity does not contribute to the execution of the executable file. The non-executable data entity is added to the executable for the sole purpose of labelling an executable file with a comment (copyright statement). The adding of comments to an executable file has no technical effect and can therefore not contribute to an inventive step.
- 3.06 The subject-matter of claim 6 does not involve an inventive step as D1 further discloses: *locating data entities using a key value by which a non-executable data entity is marked* (D1, §0053, "a new constant type is defined with a corresponding constant type tag").

**INTERNATIONAL PRELIMINARY  
REPORT ON PATENTABILITY  
(SEPARATE SHEET)**

International application No.

PCT/GB2004/000910

- 3.07 The subject-matter of claims 9-12 does not involve an inventive step as the objections made to method claims 3-6 apply *mutatis mutandis* to corresponding apparatus claims 9-12.
- 3.08 The subject-matter of claim 13 does not involve an inventive step as a computer program product for performing a method is not novel and/or not inventive if said method is not novel and/or inventive.

**Re Item VIII**

- 4 The application does not meet the requirements of Article 6 PCT, because claims 1-13 are not clear
- 4.01 The term *non-executable data entity* in claims 1-12 is not clear. It appears from the examples in the description (Description, page 8, line 8, DC LEG24 "Copyright IBM Corp. 2003") that the data entity can be a constant defining instruction (DC) and as such executable. It appears that claims 1-12 attempt to define data entities which are not used during the execution of the program. This report is drafted on that assumption.
- 4.02 The phrase *omitting the first executable data entity in dependence of the result of the comparison* in independent claims 1 and 7 fails to define the actual condition under which the first non-executable data entity is omitted. Moreover it is not clear whether the *adding of the first run unit to the executable file* is conditional as well.
- 4.03 The phrase *creating the first data entity from the located non-executable data entities* in claims 5 and 11 is not clear in that it does not define how said two non-executable data entities are combined to result in a first data entity.

M.P.Dieben  
1st Examiner

GB030030

New Page: 22 February 2005

12

CLAIMS

1. A data processing method for creating an executable file by combining a plurality of run units, the method comprising the steps of:

reading a first run unit to be added to the executable file;

locating a first non-executable data entity set to a first string value in the first run unit;

comparing the first non-executable data entity with a second non-executable data entity set to a second string value, the second data non-executable data entity being from a second run unit previously added to the executable file; and

adding the first run unit to the executable file but omitting the first non-executable data entity in dependence on the result of the comparison.

2. A method according to claim 1 wherein the step of matching matches the first non-executable data entity with the second non-executable data entity if the first string value and second string value are identical.

3. A method according to claim 1 wherein the step of comparing matches the first non-executable data entity with the second non-executable data entity if the second string value contains the first string value.

4. A method according to claim 3 further comprising the steps:

reading a third run unit to be added to the executable file, wherein the third run unit contains a third non-executable data entity of a third string value;

comparing the first non-executable data entity with the third non-executable data entity wherein a match is found if the third string value contains the first string value; and

if a match is found, removing the first non-executable data entity from the executable file, and adding the third non-executable data entity to the executable file.



GB030030

New Page: 22 February 2005

13

5. A method according to any preceding claim wherein the step of locating a first non-executable data entity comprises the steps of:

locating two or more non-executable data entities in the first run unit; and

creating the first data entity from the located non-executable data entities.

6. A method according to any preceding claim wherein the step of locating a non-executable data entity locates data entities using a key value by which a non-executable data entity is marked.

7. A data processing apparatus for creating an executable file by combining a plurality of run units, the apparatus comprising:

means for reading a first run unit to be added to the executable file;

means for locating a first non-executable data entity set to a first string value in the first run unit;

means for comparing the first non-executable data entity with a second non-executable data entity set to a second string value, the second data entity being from a second run unit previously added to the executable file; and

means for adding the first run unit to the executable file but omitting the first non-executable data entity in dependence on the result of the comparison.

8. An apparatus according to claim 7 wherein the means for comparing matches the first non-executable data entity with the second non-executable data entity if the first string value and second string value are identical.

9. An apparatus according to claim 7 or 8 wherein the means for comparing matches the first non-executable data entity with the second non-executable data entity if the second string value contains the first string value.

10. An apparatus according to claim 9 further comprising:

GB030030

New Page: 22 February 2005

14

means for reading a third run unit to be added to the executable file, wherein the third run unit contains a third non-executable data entity of a third string value;

means for comparing the first non-executable data entity with the third non-executable data entity wherein a match is found if the third string value contains the first string value;

means for removing the first non-executable data entity from the executable file and adding the third non-executable data entity to the executable file, if a match is found.

11. An apparatus according to one of claims 7 to 10 wherein the means for locating a first non-executable data entity further comprises:

means for locating two or more non-executable data entities in the first run unit; and

means for creating the first non-executable data entity from the located non-executable data entities.

12. An apparatus according to one of claims 7 to 11 wherein the means for locating a non-executable data entity locates data entities using a key value by which a non-executable data entity is marked.

13. A computer program product comprising instructions which, when executed on a data processing host, cause the data processing host to carry out a method as claimed in any one of claims 1 to 6.